



TITLE:

フローグラフの分枝数について (組合せ構造とグラフ理論)

AUTHOR(S):

夜久, 竹夫

CITATION:

夜久, 竹夫. フローグラフの分枝数について (組合せ構造とグラフ理論).
数理解析研究所講究録 1976, 259: 158-164

ISSUE DATE:

1976-01

URL:

<http://hdl.handle.net/2433/105791>

RIGHT:

フローグラフの分枝数について

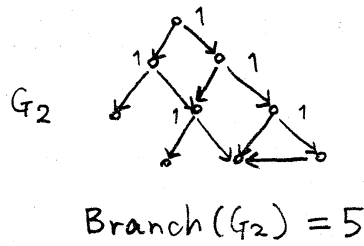
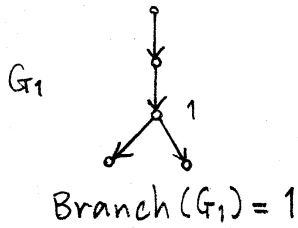
早大理工 夜久竹夫

はじめにグラフ(有向グラフ)の分枝数、
ついで生成出-樹木概念を拡張して生成
出-グラフを定義する. この小論では与え
られたグラフに対する分枝数最小の生成出
-グラフについて論じる.

グラフを $G = (N, E)$ であらわす, ここで N は node の
集合, E は edge の集合である. path, cycle, semi-
path, semicycle などの用語は Harary [1] にあづかる.
node v に対して v に入る(を去る) edge の数を 入(出)-
度数 (in (out)-degree) といい $\text{id}(G, v)$ ($\text{od}(G, v)$) と
かく.

定義. $G = (N, E)$ をグラフとする. node v の 分枝数
 $\text{Branch}(G, v)$ は (i) $\text{Branch}(G, v) = \text{od}(G, v) - 1$
($\text{od}(G, v) \neq 0$ のとき) 又 (ii) $\text{Branch}(G, v) = 0$
($\text{od}(G, v) = 0$ のとき) と定める. G の 分枝数 $\text{Branch}(G)$ は
$$\text{Branch}(G) = \sum_{v \in N} \text{Branch}(G, v).$$

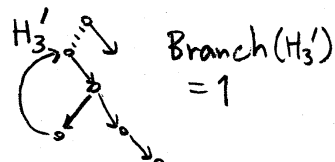
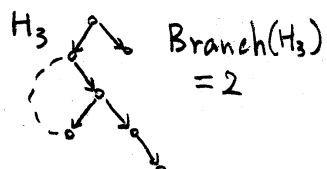
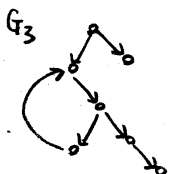
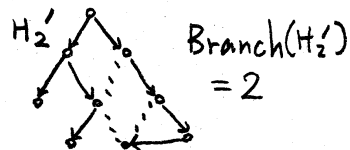
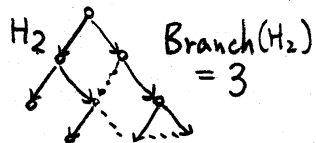
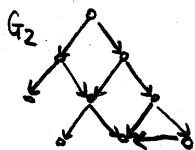
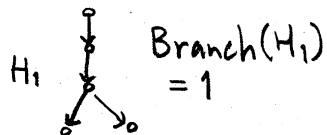
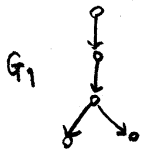
例)



生成出-樹木[1] の概念をつぎのように拡張する.

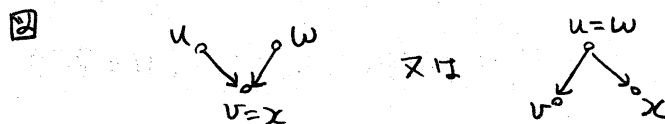
定義. グラフ $H = (N, F)$ がグラフ $G = (N, E)$ の 生成出-グラフ であるとは $\text{id}(G, v) \neq 0$ であるような全ての G の node v に対してユニークな $\text{edge}(u, v)$ ($u \in N$) が H に存在することである.

例.



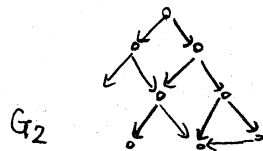
edge の間に次のような関係を導入する. この関係によるグラフの分割と、分枝数最小の生成出一樹木との関連を述べる.

定義. $G = (N, E)$ をグラフとする. G の edge (u, v) と (w, x) が alternately adjacent である $(u, v) \sim (w, x)$ とかく) とは (i) $(u, v) \neq (w, x)$ であって (ii) $u = w$ 又は $v = x$. とする.

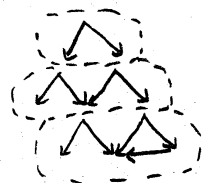


記号 \sim により \sim の反射推移関係をあらわす. グラフ $H = (N, F)$ が alternate であるとは任意の E の edge e と e' に対して $e \not\sim e'$ とする. H がグラフ $G = (N, E)$ の 極大な alternate 部分グラフであるとは F の edge と alternately adjacent な edge が $E - F$ に存在しないことである.

例



G_2 の極大な alternate 部分グラフの全て



G の semipath $p = (u_1, v_1) \dots (u_n, v_n)$ が alternate であるとは

$$(i) \quad (u_i, v_i) \sim (u_{i+1}, v_{i+1}) \quad (1 \leq i < n)$$

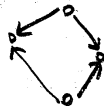
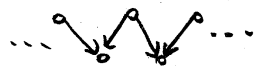
$$\text{又 (ii) } \begin{cases} u_i = u_{i+1} \Rightarrow v_{i+1} = v_{i+2} \\ v_i = v_{i+1} \Rightarrow u_{i+1} = u_{i+2} \end{cases} \quad (1 \leq i < n-1)$$

とある。又 p が alternate semicycle であるとは上の (i), (ii) に加えて $(u_1, v_1) \sim (u_n, v_n)$ であり次のいづれかが成り立つことである。

$$(iii) \quad v_1 \neq v_2 \text{ かつ } v_{n-1} \neq v_n,$$

$$(iii)' \quad u_1 \neq u_2 \text{ かつ } u_{n-1} \neq u_n.$$

例



定理 1. G_1, G_2, \dots, G_n をグラフ G の全ての極大な alternate 部分グラフとし、 H_i を G_i の生成出-グラフとする ($i \leq n$). このとき $\bigcup_{i \leq n} H_i$ は G の生成出-グラフである。

定理2. G_1, G_2, \dots, G_n をグラフ G の全ての極大な
 alternate 部分グラフとし H_i を G_i の生成出-グラフとする。
 このとき G_i の任意の生成出-グラフ H_i' に対して
 $\text{Branch}(H_i) \leq \text{Branch}(H_i')$ ならば

$$\text{Branch}\left(\bigcup_{i=1}^n H_i\right) \leq \text{Branch}(H')$$

である。ただし H' は G の任意の生成出-グラフとする。

つぎに 分枝数最小の生成出-グラフを得る アルゴリズム
 を示す。

アルゴリズム A. Input: グラフ $G=(N, E)$. ただし 出次
 数は最大 2 (70-グラフ).

Output: グラフ $H=(N, F)$.

Method: 初め $F=\phi$ とする.

While G has edges remaining do ;

if G has a node u of out-degree one
then add the unique edge (u, v) to H and
 delete all edges of the form (w, v) from G ;

else if G has alternate semicycle C
then find any edge (u, v) in C , add it to H ,
 and delete all edges of the form (w, v) from G ;
else find a node v of in-degree one, add the
 unique edge (u, v) to H and delete it from G ;

定理3. $H = (N, F)$ をアルゴリズム A により 出次数最大 2 のグラフ $G = (N, E)$ から構成されたグラフとある。
このとき H は G の 生成出-グラフで、 G の 任意の 生成出-グラフ H' に対し

$$\text{Branch}(H) \leq \text{Branch}(H').$$

G が *alternate* であるとき定理が成り立つことが証明できる。
したがって任意のグラフに対しても、定理 1, 2 より証明できる。さらに入次数も最大 2 のフローグラフについては以下が成り立つ。つぎのアルゴリズムは *semi cycle* の *searching* を必要としない。

アルゴリズム B. Input: 出次数, 入次数とも最大 2 のグラフ $G = (N, E)$, Output: グラフ H , Method: 初め $H = (N, \phi)$

While G has edges remaining do;

if G has a node u of out-degree one

then add the unique edge (u, v) to H , and delete all edges of the form (x, v) from G ;

else find any edge (u, v) in G , add it to H , and delete all edges of the form (x, v) from G ;

定理4. 出次数, 入次数とも最大2のグラフ G からアルゴリズム B により構成されるグラフを H とする。このとき H は G の生成出-グラフで G の任意の生成出-グラフ H' に対し $\text{Branch}(H) \leq \text{Branch}(H')$.

アルゴリズム A および B のオーダーはそれぞれ $O(n^3)$, $O(n)$ である, ただし n は node の数。なお, "alternate subgraph" の概念は [2] でプログラムの問題に使われている。

文献

1. F. Harary, "Graph Theory", Addison-Wesley, Reading, MA, 1968.
2. 岩田茂樹, 「goto 文最少のプログラムについて」, 本講究録, 1975.